Name: *(as it would appear on official course roster)*		
Umail address:	**@umail.ucsb.edu**	**section**
Optional: name you wish to be called if different from name above.		
Optional: name of "homework buddy" (leaving this blank signifies "I worked alone"		

1

h09

CS24 F19

h09: Heaps: Chapter 11.1 and 11.2

ready?	assigned	due	points
true	Mon 11/25 12:00AM	Mon 12/02 11:59PM	50

You may collaborate on this homework with AT MOST one person, an optional "homework buddy".

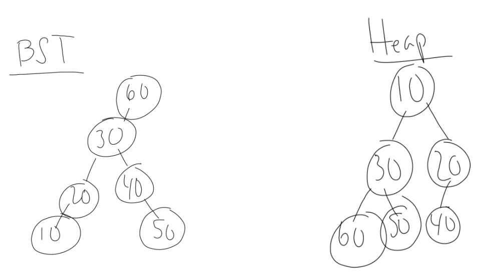
INSTEAD OF TURNING IN THIS HOMEWORK, YOU WILL TAKE A QUIZ ON GAUCHOSPACE BY THE DUE DATE.
There is NO MAKEUP for missed homework assignments.
The quiz will be made available at least two days before the due date.

Please read chapters 11.1 and 11.2 of the book on Heaps and Priorty Queues (If you don't have a copy of the textbook yet, there is one on reserve at the library under "COMP000-STAFF - Permanent Reserve").

Please:

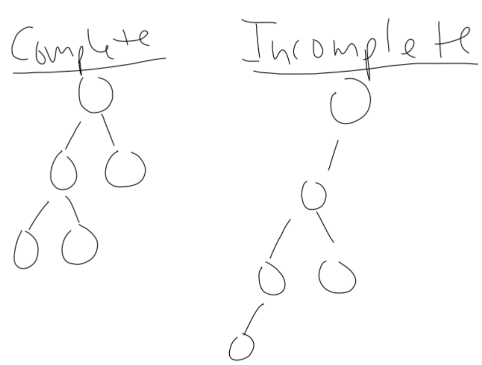
- **No Staples**.
- **No Paperclips**.
- **No folded down corners**.

1. (10 pts) Contrast a heap with a binary search tree by inserting the numbers 60, 30, 40, 50, 20, 10 first in a BST and then in a min-heap. Draw the resulting BST on the left and the heap on the right. You may draw any valid BST or Heap that contain the provided values.



2. (5 pts) In section 11.1, the book mentions that heaps are **complete** binary trees, what does that mean? Demonstrate by drawing an example of a binary tree with 5 nodes that is not complete and one that is complete.

A complete binary tree has all its levels full, except possibly for the last, and the last level is filled from left to right.



2

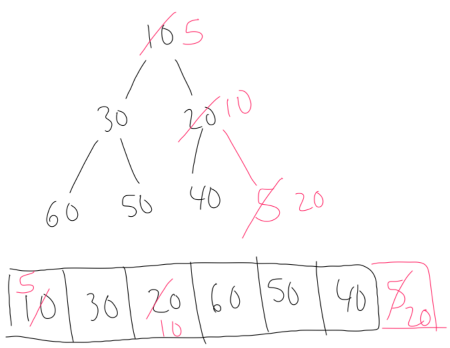
h09

CS24 F19

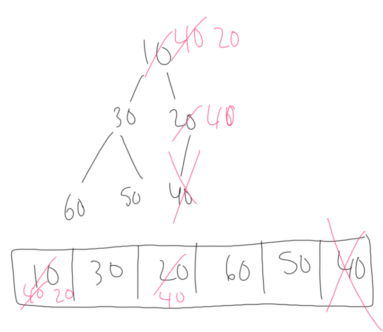
3. (5 pts) Section 11.1 mentions that complete binary trees can be implemented using arrays. Provide the array representation of the heap that you constructed in Q1 containing the key values: 60, 30, 40, 50, 20, 10

10	30	20	60	50	40

4. (10 pts) Insert the value 5 into the heap from Q3. Show how the original binary heap tree is modified on an insertion. Show also how the array representation of the heap is modfied to insert the new value.



5. (10 pts) Starting with the heap from Q3, delete the value 10 from the heap. Show how the original binary heap tree is modified on a deleting. Show also how the array representation of the heap is modfied when deleting the value.



6. (10 pts) Compare the Big -O running time of min, max, insert, delete and search for the following data structures: sorted array, generic BST, balanced BST, min-Heap, linked-list (unsorted), stack and queue

	min	max	ins.	del	search
sorted array	O(1)	O(1)	N	N	log N
gen. BST	O(N)	N	N	N	N
bal. BST	log N	log N	log N	log N	log N
min-Heap	1	—	log N	log N (for min)	—
unsorted LL	N	N	O(1) (front)	O(1) (front)	N
Stack	—	—	1 (top) only	1	—
queue	—	—	1 (front) only	1	—